



GLOSSARY OF TERMS STEMMONS ENTERPRISE

Order No	Item	Page No
1	Activity Log	4
2	API	4
3	Basic Case List vs Detailed Case List	4
4	Basic Pattern - Helpdesk	4
5	Basic Pattern - Onboarding	4
6	Basic Pattern - QA	5
7	Basic Pattern - Tracking	5
8	Cascade	5
9	Case	5
10	Cast Job - One Time	5
11	Cast Job - Recurring	5
12	Checklist - Case Launcher	6
13	Checklist - Data Capture	6
14	Checklist - Reminder	6
15	Checklist - Scorecard	6
16	Configuration Tools - Autodoc	6
17	Configuration Tools - Content	6
18	Configuration Tools - Table of Contents	7
19	Data Governance	7
20	Decode	7
21	Default Hopper	7
22	Departmental Hierarchy	7
23	Dynamic Template	7
24	Email - To Case	8
25	Email - To Entity	8
26	Entity	8
27	Entity-Entity Relationship	8
28	External Datasource	8
29	External Entity	8
30	External Entity - Ancillary Data	9
31	Facts - Standard Transformations	9
32	Facts - System Code Normalized View	9
33	Flat Table	9
34	Hopper	9
35	Inherent Metadata	10

36	Lookup Field	10
37	MTTR	10
38	Multi-Column External Datasource	10
39	Multiple External Datasource	11
40	Multiple Select	11
41	Parent-Child Cases Relationship	11
42	Person-Entity Association	11
43	Person-Role-Entity Association	11
44	System Code	12
45	System of Record	12
46	System Priority vs Case Type Priority	12
47	System Status vs Case Type Status	12
48	Taxonomy	12
49	Threshold	13
50	Trigger	13
51	Updoc	13
52	Use Case	14
53	Web Form to Case	14
54	Web Form to Entity	14
55	Work Experience Design	14
56	Workflow	15
57	XML Feed	15

1. Activity Log

An Activity Log captures specific instances of interaction with something, such as when something is created, modified, annotated, or even viewed.

2. API

Application Programming Interface is an element of a software product designed to allow interaction with other software products, so other products can execute basic commands that might typically be done by a human user.

3. Basic Case List vs Detailed Case List

A Basic Case List only shows fields that are present for and inherent in any case type, whereas a Detailed Case List shows all the specific fields associated with a single case type.

4. Basic Pattern - Helpdesk

Helpdesk is a basic Lifecycle Pattern where questions, requests, or tasks are sent by various people and someone from a specialized group is expected to respond to each request. Everyone is familiar with an IT Helpdesk, but really each department should have a Helpdesk process. This approach allows for tracking of work, accountability, effective resource allocation, great customer service (internal or external) and - more valuable than anything else - the development of a data trail that can be used to improve the organization. There can be several types of Helpdesks such as an IT Helpdesk, Marketing Helpdesk, Legal Helpdesk, Accounting Helpdesk, HR Helpdesk, not to mention things like Maintenance Tickets and Customer Service.

5. Basic Pattern - Onboarding

Onboarding is a generic pattern that captures the things that happen when something new is introduced to an organization. Regardless of the subject matter - which can be a physical thing, a customer or project, employee, or even a contract - Onboarding follows a pattern of assigning a variety of tasks to appropriate people with applicable instructions, and provides a mechanism for tracking these activities.

6. Basic Pattern - QA

Quality Assurance is a basic business process in which a subject is regularly compared to standards to identify areas of opportunities for quality

improvement. Common QA subjects include things like property or equipment inspections, but the pattern applies equally to diverse areas like employee performance, data and file quality, cyber security audits, customer service, and even employee performance reviews.

7. Basic Pattern - Tracking

Tracking is a basic Lifecycle Pattern that covers typical things associated with keeping track of things in an organization. Key elements are the ability to segment the thing by Type and Category, understand what People have Associations and Role-Associations with that thing, what Tasks or Cases are related to it, what other things connect to it, and to be able to Report on all of the above. These elements should be present regardless of what the things it - whether it is a Vehicle, Customer, Project, or a Location, effective tracking requires knowing all those things.

8. Cascade

A Cascade is a connection between two constrained or list-based data input fields where the selection of the first field trims or limits the selections available in the second field, such as selecting State would then show only applicable choices in the City field.

9. Case

A Case can be any kind of task or project, and is described both by inherent or default fields that are always present, and also by user-configured fields that are relevant to a particular type of activity.

10.Cast Job - One Time

A One Time Cast Job creates a set of Cases and assigns them to various people or hoppers based on the parameters set when the job is created.

11.Cast Job - Recurring

A Recurring Cast Job creates a set of Cases and assigns them to various people or hoppers based on the parameters set when the job is created, and continues to do so on a set schedule, re-building the recipient list each time (unless the recipients are identified by name).

12.Checklist - Case Launcher

A Case Launcher Checklist is used to create and properly assign a group of cases associated with the item or activity addressed in the form, and is particularly useful to dispatch a lot of small tasks to a disparate group of people.

13.Checklist - Data Capture

A Data Capture Checklist is used to capture specific information about a topic in a structured way, and becomes the repository for the information captured.

14.Checklist - Reminder

A Reminder Checklist functions mostly as a memory aid for a user, helping make sure that key tasks are not skipped or forgotten.

15.Checklist - Scorecard

A Scorecard Checklist is used to review the underlying subject, capture answers in a series of categories, and generate scores both, by category and overall, that can be used to improve operations.

16.Configuration Tools - Autodoc

Autodoc is an SSRS-based system deployed through the Facts Viewer that is designed to read the configuration of a selected item and output all the information needed to replicate that configuration in a different instance of Stemmons. Autodoc is helpful because - using an architecture metaphor - it essentially generates blueprints from a finished house based on the then-current state of the house. Autodoc is particularly useful in providing a comprehensive view of every part of a configuration (that might require multiple clicks if viewed in the Configuration Interface), and also as way to output a configuration in one instance, adapt or edit it for a different organization, then use a guide during configuration.

17.Configuration Tools - Content

The Content Configuration Tools helps to quickly see the End User facing parts of a configuration in an area of Stemmons, such as Case Type, Quest Form, Entity Type, Cast Job, and so forth. This is particularly helpful in getting a quick read on what is in a particular configuration; it's deeper than a Table of Contents, but not as much detail as found through Autodoc. Content may require building and deploying a Facts Report.

18.Configuration Tools - Table of Contents

The Table of Contents Configuration Tools helps to quickly build a list of available configurations in an area of Stemmons, such as Case Types, Quest Forms, Entity Types, Cast Jobs, and so forth. This is particularly helpful in looking at other instances to find Use Cases or configurations to emulate, or just having an easily accessible list of things that are possible. Table of Contents may require building and deploying a Facts Report.

19.Data Governance

Data Governance is a series of policies and processes designed to maintain data quality and create accountability for data assets.

20.Decode

A Decode is a list of possible answers to a question in Quest, along with information about what score each answer generates (if applicable).

21.Default Hopper

Once configured, a Hopper can just exist on its own, or it can be associated with one or more Case Types as the Default Hopper. When a Case is created without an Assignee (which happens sometimes if created by Email, Web Form, a misconfigured Cast Job or Quest Form, or just an End User who exits out of creating a case before selecting an Assignee), that case will automatically be assigned to the Default Hopper, which acts as an assignee of last resort so the Case does not get orphaned. Note that a Case can be manually assigned to any Hopper, not only its Default Hopper.

22.Departmental Hierarchy

A Departmental Hierarchy is a five-level taxonomy of jobs that organizes specific job titles into a series of increasingly larger groups, like Job Function, Sub department, and Department.

23.Dynamic Template

A Dynamic Template is a configuration technique that allows for multiple ways to look at a given Entity Type, with each view being dynamically generated based on the fields and formatting applicable to a given view. Dynamic Templates are

particularly useful when one Entity Type has many fields, but when different subsets of them are useful depending on the Entity Stage, Type, or Category.

24.Email - To Case

The system automatically assigns an email address to any Case Type, such that sending an email to that address will create an applicable Case.

25.Email - To Entity

The system automatically assigns an email address to any Entity Type, such that sending an email to that address will create an applicable Entity.

26.Entity

An Entity is a generic name for a discrete thing that is tracked in the system, whether it is a physical item or just a concept, and could be thought of generally as something on a list.

27.Entity-Entity Relationship

Configuring a relationship between two entities includes specifying a Parent and a Child Entity and specifying which field in each creates the link; once this is done, child entities are visible at the bottom of a parent Entity Detail page.

28.External Datasource

An External Datasource is a configuration that pulls in a list from somewhere - either in another system, or in another part of Stemmons - to populate the potential values in a field in Cases, Entities, or Quest. This allows for data normalization and is a relatively simple way to broker metadata between different business systems and processes.

29.External Entity

An External Entity is a configuration that ingests list-based data in another system to automatically create, update, and delete Entities in Stemmons, combining the benefit of information stored in an external system of record with all the benefits and processes that stem from maintaining a list of Entities in Stemmons.

Additional data fields not present in the underlying system can be added, and the resulting Entities can be configured for any process or relationship just as if Stemmons was their native system of record.

30.External Entity - Ancillary Data

Ancillary Data refers to additional fields configured in beyond those mapped into an External Entity from the underlying system of record.

31.Facts - Standard Transformations

A Standard Transformation in the context of Facts refers to additional data fields that are automatically derived from existing fields and added as additional columns in the Facts database. A classic example is the expression of a single date-time value as a variety of derivative measures, such as Year, Month, Quarter, Year-Quarter, Year-Month, and so forth. Similarly, a Person field gets transformed into easily-anticipatable ancillary data such as

32.Facts - System Code Normalized View

A System Code Normalized View in Facts refers to a Flat File containing data from multiple elements of different Types (e.g.: Different Case Types). Because elements of different Types can have very different and numerous metadata fields, the number of columns in a Flat File could balloon out of control if not limited, so Facts only sends data that is also a System Code when pushing disparate Types. However, because they are System Codes, Facts can consolidate (or Normalize) different columns into a single column with the name of the System Code. If this seems a little bewildering now, don't be surprised later if you suddenly "get it" and realize how powerful this feature actually is because it allows you to effortlessly report on things that usually have no path to combining.

33.Flat Table

A Flat Table takes information stored in one or more relational database table and formats it into columns and rows where each available piece of data is fully represented; this format makes accessing data much easier for use in reporting, algorithms, or other processes.

34.Hopper

A Hopper is a collection place for things when they are not assigned to a specific person, acting both as a holding place for work shared among a team and also as a safety net to make sure things without an assignees do not fall through the cracks.

35. Inherent Metadata

Inherent Metadata refers to characteristics of something that are always present just because of the nature of the thing itself. In Stemmons, for example, things like Cases, Entities, Quest Forms, and Cast Jobs have Inherent Metadata, which typically is built into the way the system works and is not configurable by the user. Inherent Metadata for a Case includes Created By, Assigned To, Title, Case Life, and also items that are almost always present, such as Priority and Status. Understanding Inherent Metadata is useful because those elements are always available for reporting even across disparate types of things and, unlike System Codes, Inherent Metadata does not require any configuration.

36. Lookup Field

A Lookup Field is a method of data capture where the End User types in known characters from a constrained list of items and the system returns matching values. Lookup Fields are favored where the list of possible values is long (typically populated by an External Datasource) and the End User does not need to be prompted or reminded because the desired value is somewhat known.

37. MTTR

MTTR stands for Mean Time To Repair. TTR is a calculation of the elapsed time a case is open while it is in a Status that impacts the TTR. This last part allows you to designate certain Statuses that do not accrue TTR time, such as when something is on Hold or is Inactive. The ability to configure whether time accrues during a given Status is important because time on Hold would skew the Mean if it were always counted.

38. Multi-Column External Datasource

A Multi-Column External Datasource pulls in more than one column of information for each record, allowing the End User to select from one column but passing the value from another column into the system; this is useful when a unique identifier (like Customer Number) is needed by the system for normalization purposes, but the End User needs or prefers to select from a different value (such as Customer Name).

39. Multiple External Datasource

This control allows a User to select one or more concurrent values in a field, where the values are dynamically fed by a configured External Datasource. This

approach is appropriate for situations where the list of options changes frequently (because it is easier to change these values through the standard interface) or where the options pull from a list in a different System of Record.

40. Multiple Select

This control allows a User to select one or more concurrent values in a field, where the values are entered by a Power User during Configuration. This approach is appropriate for situations where the list of options rarely changes.

41. Parent-Child Cases Relationship

As association between two cases that designates one as the parent and one as the child. This is usually used when one observed issue is caused by a deeper root cause that is being tracked in a separate case, or when a larger task spawns a number of smaller related ones.

42. Person-Entity Association

A Person-Entity Association (or just Entity Association) is a connection made between an Employee in Departments and an Entity; this Association is generic in that there is no intermediary descriptor or further classification - it's just a Person connected to an Entity. (That said, Entity Associations can be classified as Primary, in the event of multiple Associations between a Person and multiple Entities within a given Entity Type). An example would be associating a Person with a Region to show where they are based.

43. Person-Role-Entity Association

A Person-Entity-Role-Association (or just Entity Role Association) is a connection made between an Employee in Departments and an Entity, with the addition of an intermediate descriptor (the Role) that qualifies the Association based on a configured set of options. An example would be connecting a Person to a Team in the Role of Captain, or to a Customer in the Role of Relationship Manager. Entity Role Associations provide a valuable and more flexible option to not only further describe an Employee's job duties (beyond the more rigid Departmental Hierarchy), but also make connections to things like extracurricular activities and temporary assignments.

44. System Code

A System Code is a tag applied to metadata fields in Case Types, Entity Types, and Quest Forms that allows the system to understand that disparate fields actually refer to similar things. A System Code can be thought of as a higher level of metadata, or meta-metadata, which allows for normalization across many aspects of the system.

45. System of Record

A System of Record is the authoritative place where a given data element is created and maintained. In many cases, Stemmons is the System of Record, but there are often other Systems of Record such as ERP Systems, Accounting Systems, HRIS Systems, and Data Warehouses. When Stemmons connects to other Systems of Record, it tends to act as a Metadata Broker, or a Reflective System simply displaying information that lives elsewhere.

46. System Priority vs Case Type Priority

Any Case Type can include multiple values to describe the Priority of a given Case, but each of these specific status values map back to a common set of System Priorities that are numerical from 1-6. Accordingly, every individual Case has both a Case Type Priority and a System Priority, with the latter determined automatically and providing normalization for reporting and system-based actions.

47. System Status vs Case Type Status

Any Case Type can include multiple values to describe the Status of a given Case, but each of these specific status values map back to a common set of System Statuses. Accordingly, every individual Case has both a Case Type Status and a System Status, with the latter determined automatically and providing normalization for reporting and system-based actions.

48. Taxonomy

A Taxonomy is a structured classification of things such that anything can be identified and grouped easily. Children learn about taxonomies in the animal kingdom and the Dewey Decimal system. In the context of Stemmons Enterprise, Taxonomy is often used to refer to the intentional, up-front design of information architecture for the most central and important things and concepts involved in a company's business. Classic Taxonomies present in many companies include

Customers, Vendors, Products, Assets, and Documents. In many cases, Taxonomies overlap and intersect. Although the concept seems simple, designing an elegant Taxonomy can be surprisingly difficult, especially when multiple stakeholders with different responsibilities collaborate as each constituency may have its own priorities. Investing some time in basic Taxonomies is important, as many processes will be built around them, and there is an art to striking the right initial balance between simplicity and robustness. Taxonomy development is also something most people do infrequently, so it is a good place to have targeted advice.

49.Threshold

A Threshold is a pre-configured action (i.e. the creation of a Case of a certain Case Type) with various fields pre-populated based on static values, formulas/expressions, or values inherited from the associated Quest Form. A Threshold could be considered the template for the Cases created by a Quest Form, with the template selection handled by the question's Decode.

50.Trigger

A Trigger is a discretely-contained piece of software that is deployed in a Stemmons instance and is activated (or triggered) by based on the occurrence of specific, configured conditions. Triggers often are activated by things like a Case being Created, Updated, or Closed, and can contain simple lookup functions, API calls, or even complex processes like negotiating a Block chain Proof of Existence. Triggers require more technical expertise than most configuration, but they provide for incredible flexibility while still limiting customization to a discretely bound area.

51.Updoc

Updoc is a helper application that turns Stemmons Central into a metadata-driven Enterprise Content Management (ECM) or Document Management (DM) system. By providing an interface with check-in / check-out functionality, change audits, version control, security, and configurable metadata, Updoc can replace even very sophisticated ECM/DM solutions. This approach is particularly useful as Updoc is natively connected to all the configurations, enterprise metadata, and users in Stemmons, and can immediately present documents in the content of Entity Details and other Entity Relationships.

52. Use Case

A use case is a defined process of a specific business objective that needs to be accomplished by describing the combination of various external entities that exist outside the system together with system interactions needed to accomplish the business objective. The use case typically combines a variety of Stemmons tools such as Cases, Entities, Quest, Cast Jobs, Departments, and Standards. Stemmons has a library of pre-designed Use Cases that can be customized for an organization's needs.

53. Web Form to Case

Configuration of an HTML form on a website which, when filled out, generates a Case in the system. The Web Form code supports things like External Datasources and Cascades, so values available to a submitter are the same as those available to a Stemmons User. This approach often lets customers and other non-users to participate in Stemmons without even knowing it. A common example for Web Form to Case would be the creation of a Customer Support Ticket or Demo Request on a web site.

54. Web Form to Entity

Configuration of an HTML form on a website which, when filled out, generates an Entity in the system. The Web Form code supports things like External Datasources and Cascades, so values available to a submitter are the same as those available to a Stemmons User. This approach often lets customers and other non-users to participate in Stemmons without even knowing it. A common example for Web Form to Entity would be the creation of an Event RSVP or a new Sales Lead on a web site.

55. Work Experience Design

Work Experience Design is the intentional mapping of a given business process into a combination of tasks, written instructions, checklists, automations, and so forth that allows employees to do work with less training, more collaboration, and fewer mistakes. Work Experience Design contrasts with approaches that rely more on personal experience and training, and instead focuses on delivering information and instructions when- and where-needed, and on providing visibility and accountability in a productive way.

56. Workflow

Workflow is a word that means many things to different people - it can be used in a general sense to refer to the work that people do (or should do) in a given process, or in a very specific way to a self-contained series of instructions, phase-gates, if-then logic, and flow chart translated into an executable software artifact. These are very different. When describing the latter in Stemmons, we refer to Static Path Workflow, which is a classic workflow engine configured to be able to activate the Stemmons API; a qualified Power User can use SPWF to create Workflow Files that do things like create and assign cases, create entities, change values, and most other things that a human user could do. However, configuring SPWFs require more technical expertise than most configuration, are less flexible than standard approaches, and also tend to obscure processes. For these reasons, and although they can be very powerful in the right context, we tend to suggest saving SPWF for very stable processes and not leading an implementation with them.

57. XML Feed

Stemmons can automatically publish Entity activities to an XML (Extensible Markup Language) feed that can be consumed by systems like web sites, blogs, or Feed Readers. This creates a standard format information stream that can be consumed by other systems, turning Stemmons in to a basic CMS (Content Management System).